Cloud Computing for Mission Design and Operations

J. Arrieta R. Beswick D. Gerasimatos

Jet Propulsion Laboratory, California Institute of Technology

SpaceOps 2012
Stockholm, Sweden
June 12, 2012

In short

- We recognize the value of the cloud computing model, and would like to capture its benefits

In short

- We recognize the value of the cloud computing model, and would like to capture its benefits
- Valid concerns prevent widespread and expedited adoption

In short

- We recognize the value of the cloud computing model, and would like to capture its benefits
- Valid concerns prevent widespread and expedited adoption
- It is possible to expedite adoption by internally adopting the cloud computing philosophy; we propose a seven-step roadmap

A definition of cloud computing (NIST)

A model for enabling ubiquitous, convenient, on-demand access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction

A definition of cloud computing (NIST)

A model for enabling ubiquitous, convenient, on-demand access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction

- It is a model for using existing technology
- It favors the what over the how; concepts over concrete devices (e.g., storage over hard drive, or computing power over workstation)

A vision to reduce complexity

Sources of Complexity

- Proliferation of systems
- Proliferation of formats
- Proliferation of protocols
- Siloing of datasets

A vision to reduce complexity

Sources of Complexity

- Proliferation of systems
- Proliferation of formats
- Proliferation of protocols
- Siloing of datasets

Architectural Evolution

- Associative Elasticity
- Semantical Hyperdata
- Living Workflows

A vision to reduce complexity

Sources of Complexity

- Proliferation of systems
- Proliferation of formats
- Proliferation of protocols
- Siloing of datasets

Architectural Evolution

- Associative Elasticity
- Semantical Hyperdata
- Living Workflows
- **Abstraction**

A vision to reduce complexity

Sources of Complexity

- Proliferation of systems
- Proliferation of formats
- Proliferation of protocols
- Siloing of datasets

Architectural Evolution

- Associative Elasticity
- Semantical Hyperdata
- Living Workflows
- **Abstraction**

Summary of Architecture

Expose data and algorithms as resource-oriented Web services, coordinated via messaging and running on virtual machines

A vision to reduce complexity

Sources of Complexity

- Proliferation of systems
- Proliferation of formats
- Proliferation of protocols
- Siloing of datasets

Architectural Evolution

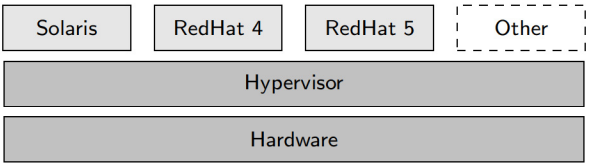
- Associative Elasticity
- Semantical Hyperdata
- Living Workflows
- **Abstraction**

Summary of Architecture

Expose data and algorithms as resource-oriented Web services, coordinated via messaging and running on virtual machines

- **Virtual Machine**
- **HTTP Server**
- **Database**
- **Message Broker**
- **Serialization Format**
- **Web Application**
- **Web Client**

Hypervisor: key to elasticity



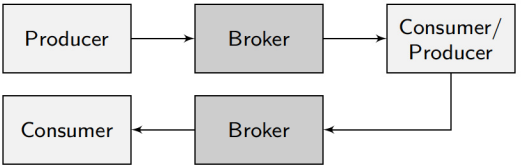
Message broker: key to abstraction

Work decoupling

Message broker: key to abstraction

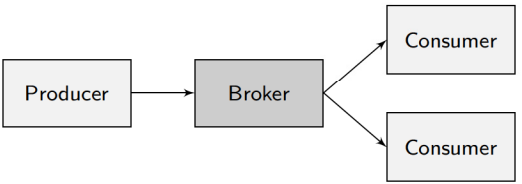
Work decoupling

Work streamlining and decoupling



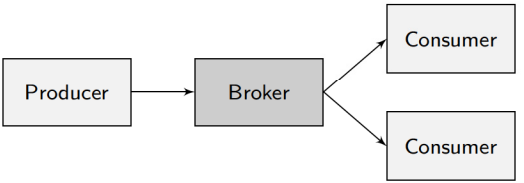
Message broker: key to abstraction

Work distribution

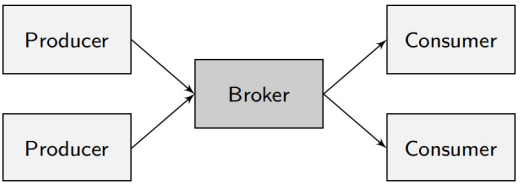


Message broker: key to abstraction

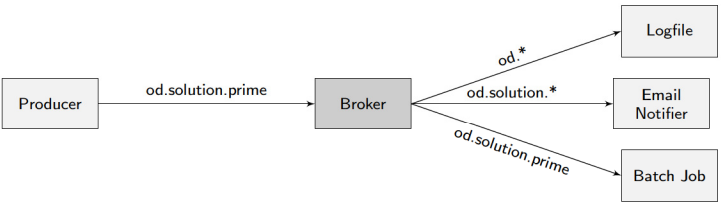
Work distribution



Work aggregation, distribution, and decoupling



Message broker: key to abstraction (example)



Data Evolution

(1) *adopt a common data-interchange format* $\rightarrow$ make data usable

typical legacy output

```
EPOCH: 18-Dec-2012 00:12:34.567891 UTC
FRAME: EME2000
POSITION: 1.234567891D+06 -7.6543210D+06 3.4567891D+02
```

Data Evolution

(1) *adopt a common data-interchange format* → make data usable

typical legacy output

```
EPOCH: 18-Dec-2012 00:12:34.567891 UTC
FRAME: EME2000
POSITION: 1.234567891D+06 -7.6543210D+06 3.4567891D+02
```

simple JSON translation

```
{"epoch":{
   "day":18, "month":12, "year":2012,
   "hour":0, "minute":12, "second":34.567891,
   "iso":"2012-12-18T00:12:34.567891",
   "clock":"utc"
   },
 "frame":"eme2000",
 "position":[1.234567891e6, -7.6543210e6, 3.4567891e2]}
```

Data Evolution

(1) *adopt a common data-interchange format* → make data usable

typical legacy output

```
EPOCH: 18-Dec-2012 00:12:34.567891 UTC
FRAME: EME2000
POSITION: 1.234567891D+06 -7.6543210D+06 3.4567891D+02
```

simple JSON translation

```
{"epoch":{
   "day":18, "month":12, "year":2012,
   "hour":0, "minute":12, "second":34.567891,
   "iso":"2012-12-18T00:12:34.567891",
   "clock":"utc"
   },
 "frame":"eme2000",
 "position":[1.234567891e6, -7.6543210e6, 3.4567891e2]}
```

(2) *evolve raw data to semantical hyperdata* → increase information content

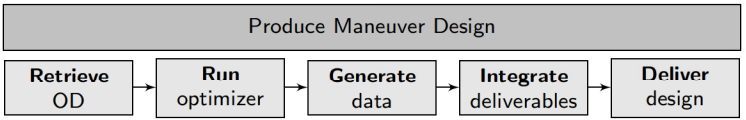
augmented JSON translation

```
{"_meta": {"_id":"c461e83ba7f549319a3e63f36649b05e",
           "product":"abcd", "mission":"mymission",
           "user":"jarrieta", "tstamp":"2012-12-18T00:11:22.334455", "sw":{"myapp", "1.2.3"},
           "sources":[{"od":"63f6bc22ef354f2c8c8252dfb6df3053"},
                      {"gin":"a53dbe90452340e89055e7c430cb0ffc"}],
           "xcheck":[{"nav":true, "rep":"jdoe", "tstamp":"2012-12-18T00:11:23.334455"},
                    {"sys":false}, {"prp":false}, {"acs":false}]},

"epoch": ...
```

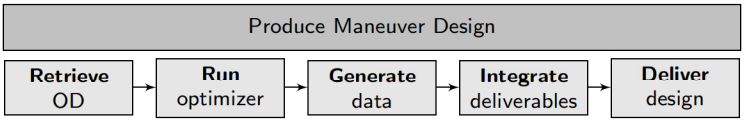
Workflow evolution

(3) *decompose workflows into simple actions* → granular, reusable units of work

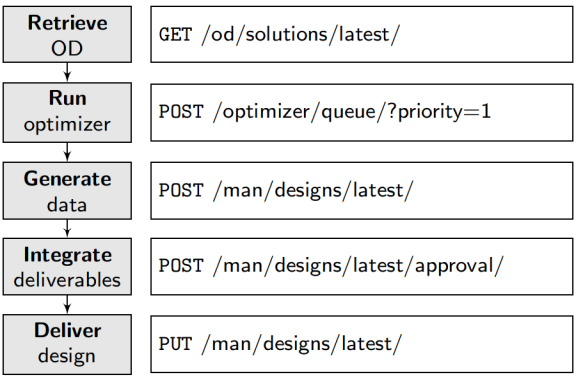


Workflow evolution

(3) *decompose workflows into simple actions* → granular, reusable units of work



(4) *categorize data and algorithms as resources* → addressable, independent entities



Protocol evolution

(5) *Provide a common interface to resources via HTTP* $\rightarrow$ access standardization

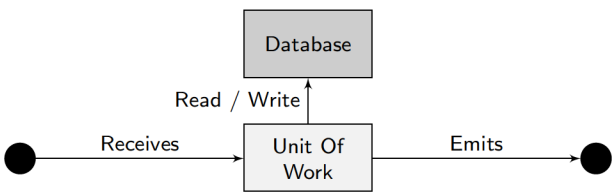
Verb	**Operation**
`HEAD`	read the metadata provided in a resource's headers
`GET`	read a resource in a specified representation
`POST`	create a resource by providing a specific representation
`PUT`	update a resource in whole or in part
`DELETE`	delete a resource

Protocol evolution

(5) *Provide a common interface to resources via HTTP* → access standardization

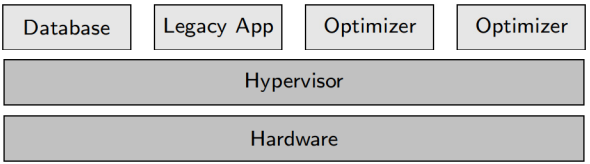
Verb	**Operation**
`HEAD`	read the metadata provided in a resource's headers
`GET`	read a resource in a specified representation
`POST`	create a resource by providing a specific representation
`PUT`	update a resource in whole or in part
`DELETE`	delete a resource

(6) *Coordinate the system interaction via messaging* → living, adaptable workflows



Protocol evolution (continued)

(7) *Deploy worker and data nodes in virtual machines* → abstract, elastic, configurable system



Conclusion

- It is possible to immediately capture some benefits of the cloud computing model
- The architecture may help reduce some common sources of complexity
- The implementation may enable teams and agencies to evaluate the cloud computing model in their specific context, with minimal infrastructure changes, and before committing to a given provider